REMARKS

Claims 1-10 are all the claims presently pending in the application. Claims 1-10 have been amended to clarify the language of the claims. Claims 11-20 have been added to claim additional features of the invention. Attached hereto is a marked-up version of the changes made to the specification and claims by the current Amendment.

It is noted that the claim amendments are made only for more clarifying the language of the claim, and not for distinguishing the invention over the prior art, narrowing the claims or for any statutory requirements of patentability. In order to avoid any reference to claim elements as steps performing a function under 35 U.S.C. § 112 Paragraph 6, amendments were made to claims 1-9 to remove the "step" language from the claims.

It is further noted that, notwithstanding any claim amendments made herein, Applicant's intent is to encompass equivalents of all claim elements, even if amended herein or later during prosecution.

Claims 1-10 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Ye (U.S. 6,374,227) in view of Ausubel (U.S. 5,905,975).

These rejections are respectfully traversed in view of the following discussion.

I. THE CLAIMED INVENTION

The present invention is directed to a rapid method for determining the set of winning bids in a combinatorial auction having multi-items and combined value bids.

The present invention uses a "column generation" technique to solve an integer programming problem formulated with a constraint matrix. A small subset of the problem is specified, its linear programming relaxation is solved, and the solution of the linear programming relaxation is used to determine which additional parts of the formulation beyond the initial subset are required. These additional parts to the original subset are generated using a column generation technique and the revised program is solved. The process is repeated until it can be mathematically proven that no additional parts of the problem need to be specified (Application,

p. 10). The resultant formulation is then solved using a commercial software package for integer programming problems.

The method, as defined exemplarily by independent claim 1, includes reading the input bidding data of items for sale, players bidding on the items, and bids that specify the player bidding, the amount bid, and the list of items included in the bid; <u>generating proposals (which comprise a collection of bids that can be awarded to a player participating in the auction) by using the input data; selecting a set of proposals such that each item is included in at most one selected proposal</u>, and informing the players bidding on the items of the selected proposals.

Conventional methods, such as those described in the background section of the application and shown in Ye and Ausubel, find an Internet auction winner by formulating the winner determination problem as an integer program and solving the program using either heuristic methods, if inexact solutions are unacceptable, or exact methods such as those implemented in commercial mathematical programming software. This approach may be adequate for many auctions of commercial interest. However, as the solution can require multiple hours, or even days on a high-performance computer, it makes rendering the conventional solutions impractical for all but the smallest instances and is most appropriate for <u>single bid auctions</u>.

The claimed method for determining winning bids, therefore, teaches a rapid solution for determining a winner for a combinatorial auction to overcome the problem of computation resources by "generating proposals," where "each said proposal comprises a collection of bids that can be awarded to a player participating in the auction," and "selecting a set of proposals such that each item is included in at most one selected proposal", as stated in Claim 1.

III. THE PRIOR ART REJECTION

The Examiner alleges that Ye would have been combined with Ausubel to form the claimed invention. Applicant submits, however, that there are elements of the claimed invention which are neither taught nor suggested by neither Ye nor Ausubel.

THE YE REFERENCE

Ye discloses a method for using the standard formulation of an integer program to formulate the winner determination problem of a multi-item auction with combined value bids. Ye uses a solver that receives an integer program and generates a linear programming (LP) relaxation solution to the integer program. An optimizer program generates an enhanced integer program that includes at least one cut according to the data of a lifted cover inequality of a specified general form that the LP relaxation solution violates (Ye col. 1, lines 60-67, col. 2, lines 1-5). Ye discloses solving the integer program with well-known methods of solving an integer program together with methods for generating a specific class of cutting planes for integer programs (such as those disclosed in commonly-assigned U.S. Pat. No. 5,216,593 by one of the present inventors) to more rapidly solve the winner determination integer program.

THE AUSUBEL REFERENCE

Ausubel discloses computer-implemented methods and apparatus for auctions, and includes discussion of multi-item auctions with combined value bids. The method compares maximized bid revenues obtained in a process iteration and decides whether to continue or stop the auction if and only if the current maximized bid revenues exceed the function of the maximized bid revenues obtained in previous iterations (Ausubel, col. 29, lines 20-28). <u>Ausubel requires that each bidder implicitly or explicitly disclose a value for every possible subset of the items</u> (Ausubel, col. 18, lines 6-17). In addition, the method determines the winners of such an auction (Ausubel, col. 20, lines 40-65 and col. 21, lines 1-8) is grossly inefficient, requiring the evaluation of $N*2^M$ single item auctions. This approach, although finite, does not scale to support auctions of commercial interest. (Application, p. 6-7).

First, applicant submits that these references would not have been combined as alleged by the Examiner. Indeed, these references are directed to different objectives and methods of performing online auctions.

Specifically, Ye is intended to find a more practical approach to finding a winner for a more complicated commercial type of combinatorial auction using a standard integer formulation

by using the cutting planes technique (Ye, col. 13, lines 45-55). In contrast, Ausubel does not use a method that supports auctions of commercial interest for combinatorial auctions and is essentially a single-bid notification system to the auctioneer to determine whether an auction should continue or end (Ausubel, col. 2, lines 21-29, col. 29, lines 22-27). Certainly no person of ordinary skill in the art would consider combining such references, absent hindsight.

Indeed, the Examiner has not cited a prior art reference that suggests in some way a modification of a particular reference or a combination with another reference in order to arrive at the claimed invention. The prior art items themselves must suggest the desirability and thus the obviousness of making the combination independent of the present invention. ACS Hospital Systems, Inc. v. Montefiore Hospital, 732 F.2d 1572, 1577; 221 USPQ 929, 933 (Fed. Cir. 1984); In re Geiger, 815 F.2d 686, 688; 2 USPQ2d 12276, 1278 (Fed. Cir. 1987) (Obviousness cannot be established by combining the teachings of the prior art to produce the claimed invention, absent some teaching, suggestion or incentive supporting the combination). Indeed, the Examiner supports the combination by merely stating that "[i]t would have been obvious ... to combine Ye's method for conducting a combinatorial auction with the teaching of Ausubel regarding maximizing auction revenue, in order to generate higher profits for the seller," which is insufficient to support the combination.

Further, even if the references were combined, the combination would not have taught or suggested each and every element of the claimed invention. Neither Ye nor Ausubel teaches or suggests "generating proposals" (which comprise a collection of bids that can be awarded to a player participating in the action) "by using input data; selecting a set of proposals such that each item is included in at most one selected proposal, and informing the players bidding on the items of the results" as recited in method claim 1 and device claim 9. Clearly, neither Ye nor Ausubel teaches or suggests these novel features.

Further, neither Ye nor Ausubel teach or suggest the claimed invention of formulating an integer program that includes a column for each proposal, a constraint for each item contained in at most one selected bid and a constraint for each player where selected bids for each player all belong to the same type, and an objective function which represents revenue, solving the integer

program for selecting the set of proposals that maximizes revenue, as recited in claim 6.

Indeed, the claimed invention does not use the methods disclosed by Ye or Ausubel. The method of the present invention uses the linear programming relaxation solution of a small subset of the problem to determine which additional parts of the formulation beyond the initial subset are required, and generates these additional parts with a column generation technique to add to the original proposal subset until the revised program is solved. (Application, p. 10).

Rather than adding additional constraints (cuts) and eliminating non-integer solutions through reducing the feasible space of the linear programming relaxation, the claimed invention uses an alternate formulation called "column generation." This technique begins with a low dimensional subset and then gradually adds further dimensional subsets which yield improvements in the solution of the corresponding linear programming relaxation of the integer program. Through careful choices in the size of the subsets, the dimensions correspond to segments of an integer solution to the original problem, and in most cases many of the issues involving finding and eliminating fractional solutions are eliminated (Application, pp. 10, 14). More specifically, claim 8 claims the column generation technique as, in addition to the elements in claim 6, solving a linear programming relaxation of the integer program for obtaining dual variables associated with each of the constraints, using dual variables for determining the excess value associated with each bid, and a threshold for each player, using a proposal generation method for selecting each player and type, a proposal for which the excess value exceeds the threshold, or determining that no such proposal exists, adding the proposals generated and repeating the steps until no new proposals are identified, and solving the integer program that includes all identified proposals. No branching or addition of cutting planes, such as the Ye method, are required to solve the winner determination problems using the claimed procedure. Neither Ye nor Ausubel disclose or suggest these methods.

In summary, Ye discloses application of the well-known integer programming technique of cutting planes to the solution of the combinatorial auction winner determination program. Ausubel discloses requires that each bidder implicitly or explicitly disclose a value for every possible subset of the items. In addition, the method determines the winners of such an auction

is grossly inefficient and does not scale to support auctions of commercial interest. Therefore, contrary to the allegations of the Examiner, Ye cannot teach or suggest the claimed invention of generating proposals and selecting a set of proposals using a different integer programming technique, column generation, to the solution of the combinatorial auction winner determination problem. Ausubel clearly fails to make up for Ye's deficiencies.

Therefore, Applicant respectfully submits that these references would not have been combined as alleged by the Examiner and even if combined, the combination would not teach or suggest each and every element of the claimed invention.

That is, turning to the exemplary claim language of independent claim 1, there is no teaching or suggestion of "generating proposals (which comprise a collection of bids that can be awarded to a player participating in the action) by using input data; selecting a set of proposals such that each item is included in at most one selected proposal, and informing the players bidding on the items of the results."

IV. FORMAL MATTERS AND CONCLUSION

In view of the foregoing, Applicant submits that claims 1-19, all the claims presently pending in the application, are patentably distinct over the prior art of record and are in condition for allowance. The Examiner is respectfully requested to withdraw her rejections and pass the above application to issue at the earliest possible time.

Should the Examiner find the application to be other than in condition for allowance, the Examiner may contact the undersigned at the local telephone number listed below to discuss any other changes deemed necessary in a telephonic or personal interview.

The Commissioner is hereby authorized to charge any deficiency in fees or to credit any overpayment in fees to Assignee's Deposit Account No. 50-0510.

Respectfully Submitted,

Date: 12/10/02

Kendal M. Sheets
Reg. No. 47,077

Sean M. McGinn
Reg. No. 34,386

McGinn & Gibb, PLLC
8321 Old Courthouse Road, Suite 200
Vienna, VA 22182-3817
(703) 761-4100
Customer No. 21254

VERSION WITH MARKINGS TO SHOW CHANGES MADE

IN THE CLAIMS:

Claims 11-20 were added. The claims were amended as follows:

1. (Amended) A method for executing a combinatorial auction, the method comprising: [the steps of]

(1) reading input data comprising:

(i) items;

(ii) player bidding on the items; and

(iii) bids, where each bid specifies the player bidding, the amount bid, and the list of items included in the bid;

(2) generating proposals by utilizing the input data, each said proposal comprising a collection of bids that can be awarded to a player participating in the auction;

(3) selecting a set of proposals such that each items is included in at most one selected proposal; and

(4) informing the players bidding on the items of the result of said selecting a set of proposals. [in step (3).]

2. (Amended) A method according to claim 1, wherein said reading input data [step (1)] comprises reading input data further including at least one type that is specified for each bid, and wherein said generating proposals [step (2)] proposals are limited to collections of bids from a player that are of the same type, and wherein said selecting a set of proposals [step (3) the set of selected proposals] is limited to sets that include at most one proposal for each player.

3. (Amended) A method according to claim 1, wherein said generating proposals [step (2)] comprises generating all possible proposals.

4. (Amended) A method according to claim 1, wherein said selecting a set of proposals [comprising enabling step (3)] is enabled by using an integer programming technique.

5. (Amended) A method according to claim 1, wherein said selecting a set of proposals [step (3)] comprises selecting a set of proposals that maximizes the total value of the bids included in the selected proposals.

6. (Amended) A method for selecting a set of bids in a combinatorial auction for at least two items involving at least one player and at least one type of bid for each player such that:

(a) each item is contained in at most one (or exactly one) selected bid;

(b) for each player, the selected bids all belong to the same type;

and among all collections of bids satisfying (a) and (b) the selected bids maximizing total revenue, said method comprising: [the steps of]

(1) generating all valid proposals;

(2) formulating an integer program that includes a column for each proposal, a constraint for each item and a constraint for each player, said constraints representing conditions (a) and (b) respectively, and an objective function which represents revenue;

(3) solving the integer program for selecting the set of proposals that maximizes revenue; and

(4) constructing a set of winning bids from the set of winning proposals.

7. (Amended) A method according to claim 6, further comprising [an additional step of] checking for ties by adding a constraint.

8. (Amended) A method for selecting a set of bids in a combinational auction for at least two

items involving at least one player and at least one type of bid for each player such that:

(a) each item is contained in at most one (or exactly one) selected bid;

(b) for each player, the selected bids all belong to the same type;

and among all collection of bids satisfying (a) and (b) the selected bids maximizes total revenue, said method comprising: [the steps of]

(1) generating a set of valid proposals;

(2) formulating an integer program that includes a column for each proposal, a constraint for each item and a constraint for each player, said constraint representing conditions (a) and (b) respectively, and an objective function which represents revenue;

(3) solving a linear programming relaxation of the integer program in said formulating an integer program [(2)] for obtaining dual variables associated with each of the constrains;

(4) using dual variables obtained in said solving a linear programming relaxation [(3)] for determining the excess value associated with each bid, and a threshold for each player;

(5) using a proposal generation method for selecting each player and type, a proposal for which the excess value exceeds the threshold, or determining that no such proposal exists;

(6) adding the proposal generated in said using a proposal generation method [(5)] and repeating said solving a linear programming relaxation, said using dual variables, and said using a proposal generation method [steps (3), (4) and (5)] until no new proposals are identified;

(7) solving the integer program that includes all identified proposals; and

(8) constructing a set of winning bids from the set of winning proposals.

9. (Amended) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for executing a combinatorial auction, said method[s] steps comprising:

(1) reading input data comprising:

(i) items;

(ii) players bidding on the items;

and (iii) bids, where each bid specifies the player bidding, the amount bid, and the list of items included in the bid;

(2) generating proposals by utilizing the input data, each said proposal comprising a

collection of bids that can be awarded to a player participating in the auction;

(3) selecting a set of proposals such that each item is included in at most one selected proposal; and

(4) informing the players bidding on the items of the results said selecting a set of proposals. [in step (3)]

10. (Amended) A computer comprising:

(1) means for reading input data comprising:

(i) item;

(ii) players bidding on the items; and

(iii) bids, where each bid specifies the player bidding, the amount bid, and the list of items included in the bid;

(2) means for generating proposals by utilizing the input data, each said proposal comprising a collection of bids that can be awarded to a player participating in the auction;

(3) means for selecting a set of proposals such that each item is included in at most one selected proposal;

(4) means for informing the players bidding on the items of the results in said means for selecting. [element (3)]

IN THE SPECIFICATION:

The specification on page 15, first full paragraph, was amended as follows:

Referring to the drawing, and more particularly to FIG. 1, there is shown a schematic diagram (10-28) of a computer implemented system for a combinatorial auction. One or more players participate in the auction. Two or more items are being auctioned. Each player uses a computer interface to interact with the auction. Each player can enter bids, and each player is informed,

through the interface of the status of his or her bids. The status of a [bed] bid is "SELECTED" if the bid is in the current set of winning bids. Having a bid "SELECTED" is the combinatorial equivalent of having the current high bid in a single item auction. That is, if no additional bids are entered, the "SELECTED" bids will become the "WINNING" bids. However, if additional bids are submitted, or if the value of an existing bid is increased, a "SELECTED" bid may become "UNSELECTED," just as in a single item auction, a current high bid may be displaced by a higher bid. Each player is allowed to edit his or her own bids, although the types of edits may be significantly restricted by the auction rules. In general, we expect that players will be allowed to increase the value of their bids. They may also be allowed to delete items from a bid (without decreasing the bid value) and to designate additional types for a bid. In some auctions, players may also be allowed to withdraw bids, perhaps subject to some penalty. This invention is not concerned with the particular bidding rules of the auction, only with the methods and systems used to select the winning bids.

The specification, beginning on page 20 line 4 and ending on page 21, line 5, was amended as follows:

Once a sufficient number of columns have been generated the objective functions *w* and the constraint matrix *A* corresponding to these proposals are constructed. The integer program is formulated. Instead of solving the integer program, we use commercial optimization software to solve only the linear programming relaxation of the integer program. This means that we relax the integrality constraints on the decision variable, and instead of requiring that each decision variable take either the value 0 or the value 1, we allow any value in the range [0,1]. Linear programs are significantly easier to solve than integer programs. Most mathematical optimization software that is capable of solving integer programs can also be used to solve the linear program. In our implementation we use the linear programming solver in OSL. Although the solution returned by the linear programming solver may specify fractional values for some of the decision variables, and thus does not correspond to a [s] set of selected proposals, it

provides valuable information in [n] the form of dual variables. There is a dual variable associated with each constraint of the integer program. If an inequality is strictly satisfied, then the dual variable associated with that constraint will have the value 0. If the constraint is strictly satisfied, then the size of the dual variable gives some indication on the relative importance of the constraint. In general, the larger the value of the dual variable, the larger the "marginal value" of the resource represented by the corresponding constraint. For the constraints corresponding to item i, we can use the dual variable π_i as an estimate of the value of the item. Using these new estimates, we can again compute the "excess" value of bids. The dual variables λ_P associated with the players also play a role in the generation of proposals. Since we can select only one proposal per player, and a large dual variable, would indicate that there are many "good" proposals for that player already being considered, we should give preference to players from [form] whom the dual variable is small. Linear programming theory says that a proposals C for player p can increase the objective function value of the linear programming relaxation of the integer program if and only if